Cartesian Control of a Cable-Driven Haptic Mechanism

Martin J.D. Otis, Vincent Duchaine, Greg Billette,
Simon Perreault, Clément Gosselin and Denis Laurendeau,
Laval University
Canada

1. Introduction

Haptic devices operated through a communication network require a trade-off between the stability of the interaction and the quality of the haptic display. A haptic device must be designed to provide the best haptic display in order to reproduce the tactile sensation of virtual objects, rigid or soft, while ensuring a stable operation to guarantee user safety. The challenges are greater when considering a locomotion interface where a walker can produce large wrenches. A Cable-Driven Locomotion Interface, used as a peripheral in a virtual environment, is designed to address some of the aforementioned issues, since the use of cables as a mechanical transmission is known to provide many advantages such as low inertia, which is helpful in attaining high speeds and high accelerations, and the potential lengths of the cables can allow for large workspaces. Using this mechanism, a walker could navigate in a virtual environment with the aid of two haptic platforms (one for each foot) which can be regarded as two independent parallel robots constrained to six degrees of freedom and sharing a common workspace.

The architecture of the framework is composed of two components: the *virtual environment manager* and the *controller manager*. The former contains the definition of the environment in which the user navigates, as expressed by a graphic rendering engine and a communication interface. The second component computes and controls the wrenches from two physical models to accurately simulate soft and rigid virtual objects. The challenge of high impact dynamics is addressed with the help of specialized reels that is also introduced as a potential solution to the issue. The aim of these new reels is to reproduce the vibrations that would normally be encountered during such an impact.

From kinematic-static duality principle, the total wrench applied on a platform is distributed optimally in each cable tension by an optimal tension distribution algorithm thereby allowing the haptic simulation of virtual objects using hybrid admittance/impedance control with multi-contact interactions. In the context of human-

robot cooperation, some practical aspects of the software design for achieving a safe control (for avoiding accidents and injuries) with a safety management plan are presented. Finally, some stability issues are also developed specifically for the cable-driven parallel mechanism.

1.1 Review

The Cable-Driven Locomotion Interface (CDLI) design presented here is based on the concept of programmable platforms with permanent foot contacts, such as Gait Master (Iwata et al., 2001), (Onuki et al., 2007) and K-Walker or the Virtual Walking Machine in (Yoon et al., 2004). CDLI employs two independent cable-driven haptic platforms constrained in six degrees of freedom (Perreault & Gosselin, 2008). Each platform is attached to a foot of the walker. Its control system and its geometry are designed so as to support a wide range of walking patterns including left/right turns and going up/down slopes or stairs that are either rigid or soft virtual surfaces or objects. In the following paragraphs, a control algorithm made specifically for cable-driven platforms is presented to address the issue of the interactions between the virtual foot models linked to the platforms and any virtual object such as but not limited to uneven terrain.

Several concepts of locomotion interfaces have been developed in order to provide a better feeling of immersion in a virtual environment and for automated walking rehabilitation. For instance, the Rehabilitation Robot LOKOMAT (Bernhardt et al., 2005) uses a hybrid force-position control method for which the force component adjusts the movement of an actuated leg orthosis so as to influence the LOKOMAT's motion and to automate user gait-pattern therapy. Such a control method is implemented in the context of the Patient-Driven Motion Reinforcement paradigm. HapticWalker is a programmable robotic footplate device that allows arbitrary foot movements during user gait training via specialized motion generation algorithms (Schmidt et al., 2005). However these control strategies are not well adapted to a CDLI as well as haptic rendering of contacts with any virtual objects or uneven terrains. In fact, a CDLI shows substantial advantages over conventional locomotion interfaces and has the potential to achieve better performances than other devices. For instance, the haptic foot platform in a CDLI can reach higher accelerations and can move in a larger workspace. Some designs involving cable-driven mechanisms were devised as the primary haptic display in a virtual environment. For instance, cable-driven devices have proven their efficiency as haptic interfaces in virtual sport training such as a tennis force display (Kawamura et al., 1995) and a catch playing simulator (Morizono et al., 1997).

In this chapter, it is shown that a hybrid admittance/impedance strategy for controlling the CDLI combines the benefits of both control classes and exploits the contact points geometry and the physical properties (stiffness, friction, etc.) of the virtual surface colliding with the virtual foot model. Within the CDLI control algorithm, the measured action wrenches imposed by the walker's feet move the platforms while a virtual reaction wrench moves the walker in the virtual environment in the event that a contact is detected between a virtual object and the virtual foot model. The software also exploits the Newton Game Dynamics™ engine, labeled "*Newton engine*" in the following, for simulating rigid body interactions.

The second section of this chapter presents the software architecture for controlling the haptic foot platform. The third and the fourth sections covers the development of the control strategy for multiple-contact points geometry that is used for performing hybrid-controlled interactions in a CDLI. The fifth one presents a custom physics engine developed under QNX OS for force rendering on a haptic foot platform so as to manage soft object

interactions. This physics engine includes a Force Optimization Problem (FOP) to distribute the wrench at each contact point uniformly and optimally. This custom engine is designed to overcome some drawbacks of the Newton engine, such as transient force computation and object penetration that occurs when a contact is detected between the virtual foot model and a compliant surface. Finally, the last section of the chapter presents simulations of the control strategy with the physics engines in normal gait walking conditions.

1.2 The geometry of the CDLI

As shown in figure 1, the geometry of the CDLI is optimized to cover the largest workspace possible in a limited volume (i.e. the overall dimension of the complete CDLI) so as to avoid cable interferences and to minimize human-cable interferences while the user is walking (Perreault & Gosselin, 2008). It must be noted that due to the unilaterality of the actuation principle, a cable-driven parallel platform needs at least seven cables in order to control a six DOF platform. Since each platform has six DOF so as to emulate human gait (Yoon & Ryu, 2006) and all cable attachment points are chosen so as to reach an optimal workspace, each haptic foot platform is actuated by eight cables.

The dimensions of the workspace along the X, Y and Z axis are respectively 2 metres, 0.6 metre and 1 metre, all within the overall dimensions of the complete CDLI whose size is approximately 6.0 metres by 3.5 metres by 3.0 metres. These dimensions allow users to perform a wide range of walking patterns.

The model of the virtual foot in the virtual environment, shown in figure 2, is mathematically related to the haptic foot platform by a translation vector and a rotation matrix between their respective reference frames.
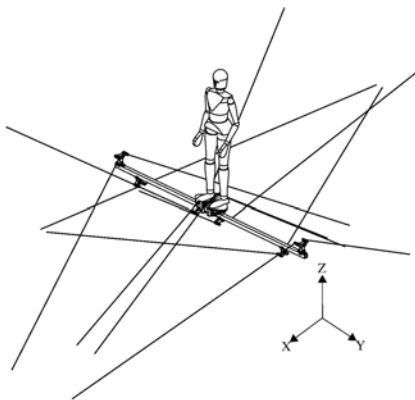

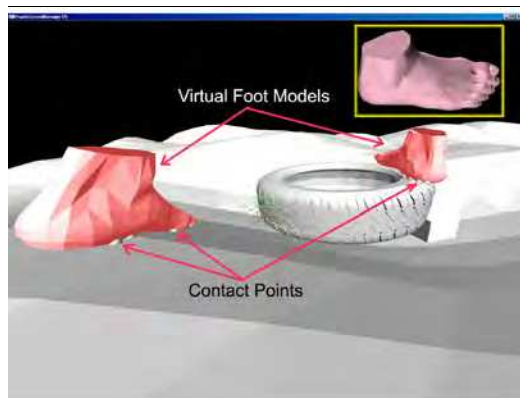Fig. 1. CAD model of the complete CDLI taken from (Perreault & Gosselin, 2008)

Fig. 2. Virtual foot models in contact with virtual objects

1.3 Control Classes for Haptic Rendering

Two control classes are generally employed for haptic rendering on the platforms: an impedance control class and an admittance control class similar to those described in (Carignan & Cleary, 2000). Since both control classes use pose and wrench inputs, they are

instead defined by the output or the feedback loop. The properties of each approach are compared in table 1. The Cobotic Hand Controller (Faulring et al., 2007) and the HapticMaster (van der Linde & Lammertse, 2003) are both mechanisms that use admittance control. On the other hand, the Excalibur (Adams et al., 2000) and the Phantom (McJunkin & al., 2005) haptic devices have been designed as impedance displays that use impedance control.

Indeed, two virtual object models could be defined: an admittance model and an impedance model. Using linear circuit theory (quadripole or two-ports models), there are four possible topologies described by the immitance matrices: the impedance matrix, the admittance matrix, the hybrid matrix and the alternate hybrid matrix that the controller could manage as described in (Adams & Hannaford, 1999).

The hybrid control strategy combining these two control classes (interacting with the both virtual object models) ensure that free movements and contact with a rigid virtual object are rendered realistically by the platforms. Section 4 describes a method for selecting the appropriate control class using both the geometry of the contact points and the virtual object properties.

Impedance-controlled system (impedance control with force feedback)	Admittance-controlled system (admittance control with position/velocity feedback)
Controls the wrench applied by the haptic foot platform	Controls the pose or velocity of the haptic device
Is subject to instabilities when the user releases the device	Is subject to instabilities when the stiffness of the user's legs increases
Can simulate highly compliant virtual objects	Can simulate an unyielding virtual object

Table 1. Comparison of the control classes

1.4 Stability issues

The capability for a human to stabilize an unstable system or even to destabilize a stable system is a recurrent problem in the haptic interface control. Two methods are developed in the literature for stability analysis. The first one is based on human and environment models (on-line or real-time computation of the muscle stiffness) in order to adjust an admittance model in the controller that gives pose setpoints computed from the user applied wrench measured at the end effector. This method consists in adjusting the control law for ensuring stability of the system (Tsumugiwa et al., 2002). The analysis of the stability could then be performed with different strategies such as Routh-Hurwitz, root-locus, Nyquist, Lyapunov or μ-analysis among others. In the other case, the second method does not use any model. This method analyses the transfer of energy inside the system like in (Hannaford & Ryu, 2002). On the other hand, there exist numerous stabilizing techniques such as those exploited in adaptive or robust control.

A stable haptic system dissipates more energy than the overall control system produces. However, this diminishes the realism of the haptic display as the dissipated energy

increases. It is therefore a trade-off between performance and transparency. In cable tension control applications the dissipated energy should be compensated for so as to lead the system toward an unstable regime. The stabilizing method uses a virtual damping parameter in order to dissipate accumulated energy with a passivity observer (PO) and a passivity controller (PC). This method was used also for compensating the delay on the network.

Friction hysteresis in reel increases vibrations in the cables when the reel's mechanical parts stick and slip. Furthermore, rigid contacts between the virtual object and the foot produce discontinuities in cable tensions that have a tendency to create or emphasize cable vibrations. Finally, the stiffness of the reel and of the mechanical structure should be at least larger than the one of the virtual object so that mechanical deformation cannot generate more instability. From this analysis, which excludes the electronic hardware, six types of instability inside a hybrid control architecture for a Cable-Driven Mechanism can be considered:

1.   Cable vibration and tension discontinuities;
2.   Mechanical design (stiffness of the overall mechanical structure including motorized reel, friction hysteresis, actuator dynamic, encoder resolution, etc.);
3.   Hybrid control architecture with uncertainty (Cheah et al., 2003) and with flexible joint (Goldsmith et al., 1999);
4.   Contacts with a stiff virtual object with one or more contact points (Lu & Song, 2008);
5.   Interaction between a human and a mechanism (Duchaine & Gosselin, 2009) and
6.   Time delay (latency) over the network (Changhyun et al., 2008).

2. Software Architecture for Control

The hardware architecture is composed of two components: a soft real-time module implemented on a standard PC running Windows which manages the virtual environment with a graphic rendering engine, and a hard real-time module implemented on a standard PC running QNX whose primary tasks is to control and drive the cable-driven platforms and a server that ensures intercommunication and synchronization between different walkers. The software architecture is designed to exploit the above hardware and is thus composed of the two components shown in figure 3: the *Virtual Environment Manager* and the *Controller Manager* which are described in the next sections.

2.1 Virtual Environment Manager

The *Virtual Environment Manager* (VEM) is responsible for handling haptic objects (virtual foot model and virtual object), a physics engine, and a virtual user (an avatar) whose feet are shown to be moving in a virtual environment. The avatar therefore mimics the movements of the user so that he or she can observe his actions in the virtual environment. The *virtual user* defines the characteristics of the walker who can observe the virtual environment in coherence with his feet. For the *physics engine*, Newton Game Dynamics™ is used as a slave engine while the master physics engine is implemented on a second PC using QNX OS in the *controller manager* as described in section 2.2. The communication between both physics engines is ensured by a *client communication interface* and a *server communication interface*.

The *Haptic Scene Manager* (HSM) is the main interface with which the virtual environment is built and configured. The HSM is responsible for configuring the Newton engine according to the simulation requirements.
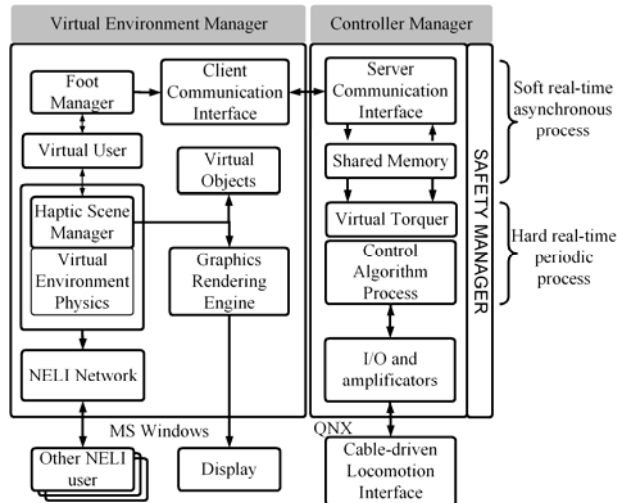

Fig. 3. Software architecture

It is also responsible for the creation, set up, and destruction of virtual objects having a haptic presence in the environment. Besides the HSM, the haptic module uses two other managers for the hands (*hand manager*) and feet (*foot manager* that define the avatar). The *foot manager*, which is connected to the *virtual user*, communicates with the *controller manager* using a TCP/IP connection. Over this communication link, the Newton engine provides the contact points between each virtual foot model and the virtual object to the *controller manager* and also provides the normal/tangent vectors to these contact points as well as the penetration into the virtual object. Conversely, the *controller manager* responds to these inputs by providing the *foot manager* with the pose and the speed of the haptic foot platform resulting from the contact, as well as the total wrench computed by the custom physics engine which then moves the virtual foot model and the virtual object in the scene.

The communication link between the VEM and the *controller manager* must support a minimum transmission rate of approximately 100 Hz in order to transfer a burst of 512 bytes with a maximum latency of one millisecond. Although there are hardware solutions satisfying these requirements, the main issue still remains the latency of the asynchronous process which is only executed whenever possible. Some solutions for resolving communication bandwidth limitations are given in (Sakr et al., 2009), where a prediction approach is exploited with the knowledge of human haptic perception (Just Noticeable Differences of Weber's law). The definition of a deadband is used for data reduction. This deadband consists of velocity and pose threshold values where there are no significant new informations. In the proposed system described in this chapter, the quantity of data transmitted over the network is based on the selection of meaningful contact points from those evaluated by the Newton engine. In fact, only three points are required by the *controller manager* to define the control class that will be applied in the appropriated DOF.

2.2 Controller Manager

The *controller manager* runs two processes: a hard real-time periodic process (labeled *control algorithm process*) responsible for the hybrid control algorithm, and a soft real-time asynchronous process that manages the virtual environment updates between the *foot manager* and the control algorithm process. The periodic process can be pre-empted any time by the asynchronous process. The rate of the periodic process for controlling the actuators and the sampling rate achieved for wrench sensing are both set at a multiple of the analog input signal number and has a minimal rate of 500 Hz, and in the best case, 1 kHz.

The *virtual torquer* in tandem with the control algorithm process runs the master physics engine (labeled *Haptic Display Rendering* (HDR) in figure 4) as well as a washout filter that maintains the walker at the centre of the workspace using a variable impedance model and position feedback as described in (Yoon & Ryu, 2006) and (Yoon. & Ryu, 2009).
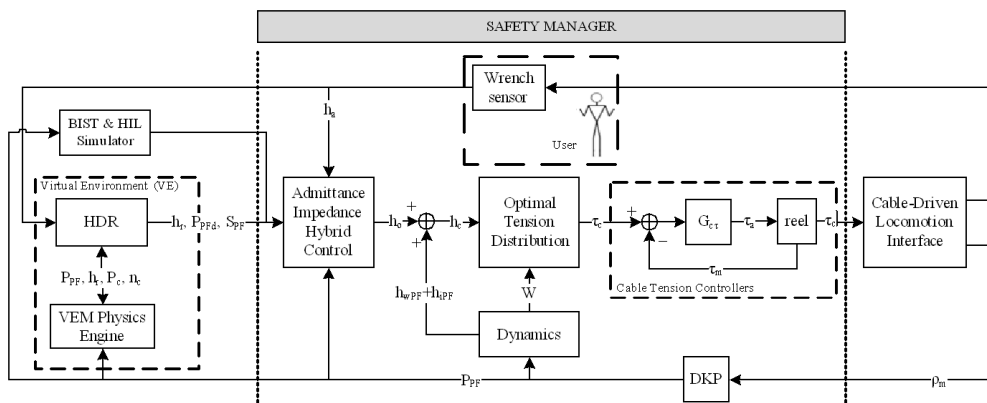

Fig. 4. Simplified control algorithm process with interactions of both physics engines

The control algorithm process, detailed in figure 4, accepts one input (the output of a 6 DOF wrench sensor) and produces two outputs (cable tensions τ_c and platform poses $\mathbf{P}_{PF} \in \mathbb{R}^6$). The appropriate reaction wrench $\mathbf{h}_r \in \mathbb{R}^6$ is computed from the interaction between both physics engines. These engines also determine whether the degrees of freedom for each platform should be controlled in impedance or in admittance. Depending on the selected control class, the 6 DOF wrench sensor can produce, an action wrench $\mathbf{h}_a \in \mathbb{R}^6$ which moves each platform using a hybrid control scheme.

The total wrench $\mathbf{h}_c \in \mathbb{R}^6$ applied at the centre of mass of the platform is balanced with positive cable tensions using an Optimal Tension Distribution (OTD) algorithm as described in (Fang et al., 2004). The result being a set of equilibrium tension values τ_c, called the *setpoint*, that the cable tension controllers then attempt to follow. The pose of each platform is computed with the Direct Kinematic Problem (DKP) algorithm using the lengths of the cables ρ_m as input.

Since a virtual object can be rigid or soft, two physics engines are implemented to ensure a general approach that allows the physical reactions between the platforms and virtual objects to be adjusted. The HDR decides which reaction wrench computed by both engines must be transferred to the hybrid control. This choice depends both on the properties of the

virtual object and on the contact points geometry. The contact point detection and the associated normal vector at the interface between a virtual object and a virtual foot model is evaluated by the Newton engine and dynamic proxy objects. The HDR exploits these values to compute its own reaction wrench $\mathbf{h}_r$ and for selecting which control class to use in order to get the best haptic rendering.

2.3 Cartesian Compensations

Mechanism transparency is crucial when a walker has to use a mechanical device inside a virtual environment. Indeed, in the virtual world, the user must be able to forget the fact that he is attached and that he is using a real device. Only the simulated physics (such as friction between foot and virtual object) inside the virtual environment must be reproduced under the user's foot. In order for this to happen, it is very important to know the exact behaviour of the mechanism at any time. This is made possible by knowing the dynamics of the device.

In a locomotion interface, the inertia and weight of platforms and sensors must be compensated for in order to increase the realism of the haptic display to the user. Therefore, $\mathbf{h}_c$ not only includes the variable load $\mathbf{h}_a$ applied by a walker's foot on the platform and the set of wrenches $\mathbf{h}_r$ computed from the interaction between walker's feet and its virtual environment, but also the effect of the weight $\mathbf{h}_{wPF} \in \mathbb{R}^6$ and inertia $\mathbf{h}_{iPF} \in \mathbb{R}^6$ of the platform and wrench sensors. For impedance control with force feedback, an additional $\mathbf{h}_r$ is added for haptic rendering of virtual contact between the platform and the virtual object.
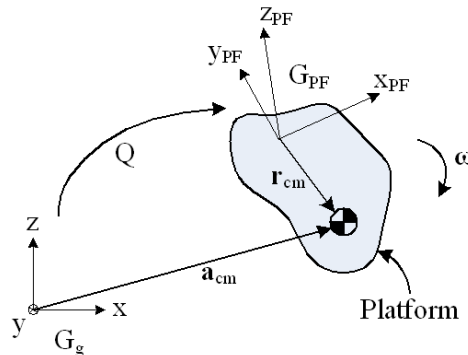

Fig. 5. Reference frame of the platform

The compensation for the mechanism inertia and weight (platforms and sensors altogether) is computed by dynamic wrenches $\mathbf{h}_{iPF}$ and $\mathbf{h}_{wPF}$ respectively. Since there are two working frames, the inertial frame G_g and the moving frame attached to the end-effector G_{PF} (as described in figure 5), and no deformation is permitted to the platform, $\mathbf{h}_{iPF}$ can be defined as follows:

$$\mathbf{h}_{iPF} = - \begin{bmatrix} m\mathbf{a}_{cm} \\ \mathbf{Q}[\mathbf{I}_{cm}]\mathbf{Q}^T\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{Q}[\mathbf{I}_{cm}]\mathbf{Q}^T\boldsymbol{\omega} \end{bmatrix},$$
$$\mathbf{a}_{cm} = \mathbf{a} + \dot{\boldsymbol{\omega}} \times \mathbf{Q}\mathbf{r}_{cm} + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{Q}\mathbf{r}_{cm}),$$

(1)

where the scalar noted m represents the mass of the platform, the vector noted $\mathbf{a}_{cm}$ represents the acceleration vector of the centre of mass of the platform in the inertial frame (i.e. the global reference frame), $\mathbf{I}_{cm} \in \mathbb{R}^{6 \times 6}$ is the inertia matrix of the platform to its centre of mass and defined in the mobile frame G_{PF} (this matrix is constant since the mobile frame is fixed to the platform), $\boldsymbol{\omega}$ is the angular velocity vector of the moving frame G_{PF} compared to the inertial frame G_g, and $\mathbf{r}_{cm}$ is the vector connecting the origin of the moving frame to the centre of mass of the platform in G_{PF}.

The value of $\mathbf{h}_{iPF}$ is negative since it removes the inertia of the moving mechanism. Also the evaluation of $\mathbf{a}_{cm}$ with a low level of noise could be difficult with a low resolution of quadrature encoder inside the reel. This value should be evaluated with a six axis accelerometer/gyroscope module installed near the centre of mass. For the system presented in this chapter, it is not recommended to evaluate $\mathbf{a}_{cm}$ with the wrench sensor since the wrench sensor is used in the hybrid control.

Finally, to complete the part of dynamic relations related to the platform of the mechanism, it is needed to describe the wrench of the weight of the platform $\mathbf{h}_{wPF}$. Thus, this relation is defined as follows:

$$\mathbf{h}_{wPF} = \begin{bmatrix} m\mathbf{g} \\ \mathbf{Q}\mathbf{r}_{cm} \times m\mathbf{g} \end{bmatrix}, \tag{2}$$

where the vector $\mathbf{g}$ is the gravitational acceleration vector. As for the inertia of the motors and reels, they are accounted for by the cable tension controllers which also consider the effects of friction at low speed in order to accelerate the responses of their respective control loop.

2.4 Optimal Tension Distribution

Since each platform is driven by n-6 redundant cables, it is important that the tension be distributed among them according to kinematic and dynamic conditions so as to minimize the actuation power over all actuators (Hassan & Khajepour, 2008). It is desired to maintain the tension in the cables above a minimum threshold value τ_{min} to limit cable sagging. Such a threshold must be greater than the minimal tension set by the precision of the acquisition system combined with a performance criterion obtained from cable behaviour (Otis et al., 2009a). Actuators (i.e. reel, motor and cable) are also limited by a maximum torque τ_{max} which helps to avoid control problems. Hence, the following force distribution method is proposed to avoid cable sagging as well as excessive mechanical deformation of the CDLI:

$$\begin{cases} \text{minimize} & \frac{1}{2}\boldsymbol{\tau}^T \mathbf{G} \boldsymbol{\tau} \\ & \mathbf{W}\boldsymbol{\tau} = \mathbf{h}_c \text{ , with} \\ \text{under} & \tau_{max} \geq \tau_i \geq \tau_{min} \\ & \text{if interference, } \tau_{max} = \tau_{min} \end{cases} \tag{3}$$

$$
\begin{aligned}
\mathbf{G} &= \mathrm{diag}(g_i) \\
\mathbf{h}_c &= \mathbf{h}_{wPF} + \mathbf{h}_{iPF} + \mathbf{h}_o \\
\boldsymbol{\tau} &= \begin{bmatrix} \tau_1 & \tau_2 & \ldots & \tau_n \end{bmatrix}^T \\
i &= 1,\, 2,\, \ldots,\, n \text{ cables,}
\end{aligned}
\tag{4}
$$

where $\mathbf{h}_c$ represents the forces and torques that are applied on a single platform (i.e. the wrench applied by the cables on that platform), τ_i is the tension vector of the ith (of n) cable, $\mathbf{W}$ is the pose-dependent wrench matrix computed by the platform Jacobian matrix that links Cartesian to articular velocities, $\mathbf{G}$ is a weighting matrix with its diagonal elements such that $g_i = 1$ for all i, where the mathematical derivation of (3) is presented in (Barrette & Gosselin, 2005) and an application is described in (Perreault & Gosselin, 2008).

2.5 Human safety and security management plan
In the context of a human and a mechanism interacting within the same workspace, safety for human user is one of the utmost importance issues to be considered for avoiding accidents and injuries. The overall control algorithm process has a safety manager with an error handler that was designed with the help of a risk study. Each component of the software must have self-testing capabilities (or BIST for Build-In Self Test) for a general system test planning for the purpose of quality assurance (QA) and safety management. A Hardware-in-the-loop (HIL) simulator could be implemented as a way for running some parts of the BIST and partially control the platform. Documentations can be found in the IEEE 829 Standard for Software Test Documentation, CSA Z432-04 and ISO 14121. For Cable-Driven Mechanism applied to haptic applications, a minimum of four safety issues must be considered and documented:

1. Sensors reliability or fault tolerant (cable cut or failure by fatigue);
2. Mechanical interference like cable interference and platform interference with other parts of the mechanism or the user (Otis et al., 2009a);
3. Workspace limitations when the platform is going outside of its workspace;
4. Human and robot interaction like :
   - The mechanical device that safely disconnects the user from the mechanism when the mechanism is out of control (Lauzier & Gosselin, 2009) and,
   - The safety tether which maintains the equilibrium of the user when walking, falling or when the mechanism is out of control (Ottaviano et al., 2008), (Grow & Hollerbach, 2006).

Other safety aspects of the system must also be guaranteed. For example, the system must manage any sensor destruction and limits on control values (cable length, maximal and minimal cable tension, maximal current send to the motor, maximum wrench generated from the physics engine, etc.). Finally, a watchdog timer is included to ensure that the control algorithm process is executed within the prescribed period of the periodic process within an error of 5%. This watchdog and the timing period are set using a hardware interrupt implemented on a data acquisition board that is independent from the software to avoid control failure and to ensure hard real-time response. For computing derivative and for reducing noise on this value, the algorithm should consider the time shift generated by

the latency (the 5% error on the prescribed period) of the OS context switching (or other process running).

3. Admittance/Impedance/Inertial-Wrench Hybrid Control

Hybrid control is a general approach that exhibits the combined advantages of impedance, admittance, and inertial-wrench control (or more precisely a null wrench control). The structure of the admittance/impedance hybrid control for one platform is shown in figure 4 and is detailed in figure 6. Two identical control structures are implemented, one per platform. The selection of the control class for each DOF of the platform is achieved by the Π $\in \mathbb{R}^{6\times6}$ matrix. The state of the Π matrix depends on the orientation of contact points geometry and the orientations of the platform.

When the reaction force $\mathbf{h}_r$ is null and the impedance control class is selected by the Π matrix, one simply chooses a null force control scheme with an open gain loop G_{ch}=K. Otherwise, impedance or admittance control is applied on the desired DOF for each platform. Admittance control could be performed by velocity or position feedback which could produce different experimental results, as described in (Duchaine & Gosselin, 2007). The desired platform positions $\mathbf{P}_{PFd}$ (or the desired velocities) are defined by the contact points given by the Newton engine. As the strategy used by the Newton engine, a wrench $\mathbf{h}_p$ must be added to the admittance control to avoid any large penetration inside a virtual object when a collision detection may have been missed because the refresh rate is not performed in time. This strategy also avoids the computation of a new set of contact points as the foot enters the object. In the Newton engine, the wrench $\mathbf{h}_p$ is computed with an impedance model of the object and must be controlled in the physics engine since the command is a null penetration for a rigid contact. From figure 6, the wrench $\mathbf{T}^{-I}_{cm}\mathbf{h}_o$ to be computed by the hybrid controller is defined by equations (5) to (8) :

$$\begin{aligned} \mathbf{T}^{-I}{}_{cm}\mathbf{h}_o = \Pi(G_{cp}(\mathbf{P}_{PFd}-\mathbf{P}_{PF})+\mathbf{h}_p)+ \\ (\mathbf{I}-\Pi)(G_{ch}(\mathbf{h}_r-\mathbf{h}_a)+\mathbf{h}_r)-\mathbf{K}_h\mathbf{h}_a \text{ with,} \end{aligned} \tag{5}$$

$$\Pi = \mathbf{Q}_o\mathbf{Q}_g\mathbf{S}_c\mathbf{Q}_g^T\mathbf{Q}_o^T, \tag{6}$$

$$\mathbf{Q}_o = \begin{bmatrix} \mathbf{Q} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{Q} \end{bmatrix}, \tag{7}$$

$$\mathbf{Q}_g = \begin{bmatrix} \mathbf{Q}_c & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{Q}_c \end{bmatrix}, \tag{8}$$

where $\mathbf{G}_{cp}$ is a standard filter that controls the desired position $\mathbf{P}_{PFd}$ (or the desired velocity) of the platform ($\mathbf{P}_{PF}$ is the measured position), $\mathbf{Q}_c \in \mathbb{R}^{3\times3}$ is the rotation matrix between the contact points reference frame G_c and the platform reference frame G_{PF}. $\mathbf{Q} \in \mathbb{R}^{3\times3}$ is the rotation matrix between reference frame G_{PF} and its global counterpart G_g, which is computed by the DKP with the cable lengths ρ_m. $\mathbf{G}_{ch}$ is the wrench controller which should

be set high enough (bounded by the appropriate stability criteria) to reduce the errors caused by the dynamics and friction of the cable-driven platform and of the motorized reels. A transfer matrix $\mathbf{T}_{cm}$ is used for computing the output wrench at the centre of mass of the platform since all haptic wrenches are under the foot and the OTD uses the centre of mass as its reference. Also, to prevent the platform form sticking to the contact point (i.e. when the hybrid control is oscillating between admittance and impedance), the action wrench $\mathbf{h}_a$ is added to the output of the hybrid controller with a gain $\mathbf{K}_h$. This gain and the two Cartesian controllers must consider the geometry of the mechanism and stability margins. In a Cable-Driven Mechanism, an anisotropy geometry could be designed and the control would need more energy in some DOF than other for obtaining the same transparency. Note that the initial conditions of the integrators and the filters inside both $\mathbf{G}_{ch}$ and $\mathbf{G}_{cp}$ must be adjusted for avoiding bouncing and instability. Furthermore, in some circumstances, kinematics and dynamics uncertainties must be considered in a hybrid control as described in (Cheah et al., 2003).
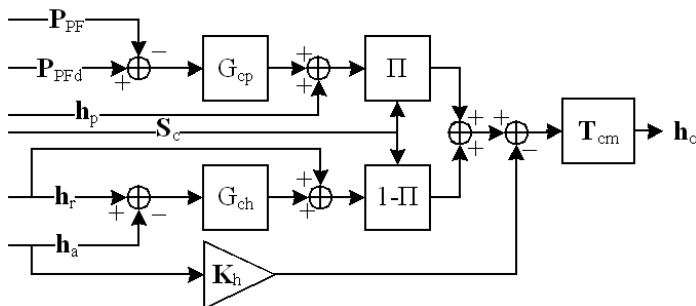
Fig. 6. Admittance/Impedance/Null Force Hybrid Control

The selection between control classes is achieved by the diagonal selection matrix $\mathbf{S}_c \in \mathbb{R}^{6 \times 6}$ (1 or 0 on the diagonal and other values are set to 0) and is evaluated in the contact point reference frame G_c. The values on the diagonal of matrix $\mathbf{S}_c$ depend on friction, contact points geometry, and calibration based on experiments. A second selection matrix, $\mathbf{\Pi}_o$, defined in equation (9), is used to compute the force at each contact point by selecting the DOF under constraint using the *Force Optimization Problem* (FOP) algorithm defined in section 5.2:

$$\mathbf{\Pi}_o = \mathbf{Q}_o \mathbf{Q}_g \mathbf{S}_o \mathbf{Q}_g^T \mathbf{Q}_o^T \qquad (9)$$

Thus, a 0 on the diagonal of matrix $\mathbf{S}_o \in \mathbb{R}^{6 \times 6}$ allows a null force control by providing a corresponding null component for the wrench in contact points reference frame G_c. These two selection matrices ($\mathbf{S}_c$ and $\mathbf{S}_o$) are thereby quite similar in function, albeit not identical.

4. Definition of the multi-contact points geometry

Since the control strategy exploits two physics engines (Newton engine and HDR), each engine can control a given platform's DOF either in admittance or in impedance simultaneously. The virtual object properties and the contact points geometry are the criteria that determine the appropriate control class using the selection matrix Π that satisfies the following properties:

1. For a collision between a virtual foot model and a rigid virtual object for which a friction model is implemented, the selection could be based on the geometry described by the contact points between the virtual object and the virtual foot model;

2. To simulate friction, impedance control with force feedback could be chosen because there is a tangent force at the contact point reacting to an applied force from the user;

3. For compliant virtual objects, impedance control could be chosen and

4. Movement in free space could be simulated by a null force control, a special case of impedance control when some components of $\mathbf{h}_r$ are equal to 0.

The favoured method used for selecting a given control class is a multi-contact points strategy (shown in figure 7) that emphasizes control advantages relating to the simulation of rigid virtual objects which includes a friction model. Contact points are computed as the minimum set of points that completely define the boundary of the intersection between a virtual foot model and a given virtual object. They are represented in the Newton engine in conjunction with a corresponding set of normal vectors. For a haptic foot platform, a set of points whose relative distances are within ten millimetres can be viewed by the control algorithm as a single point.

The multi-contat points strategy used in this case involves the direction of the user-applied wrench for each foot: if a component of the measured wrench $\mathbf{h}_a$ is in the same direction as a normal vector describing contact points geometry, which means that the user pushes on the virtual object, this direction (or DOF) is then constrained by admittance control for rigid virtual objects; otherwise either null force control is selected to simulate free movement (i.e. the contact point is eliminated) or impedance control is employed to simulate friction. In the case of a soft virtual object, impedance control is selected in the direction normal to the contact points geometry. In figure 7, the normal vector describing contact points geometry is along the z_c axis.
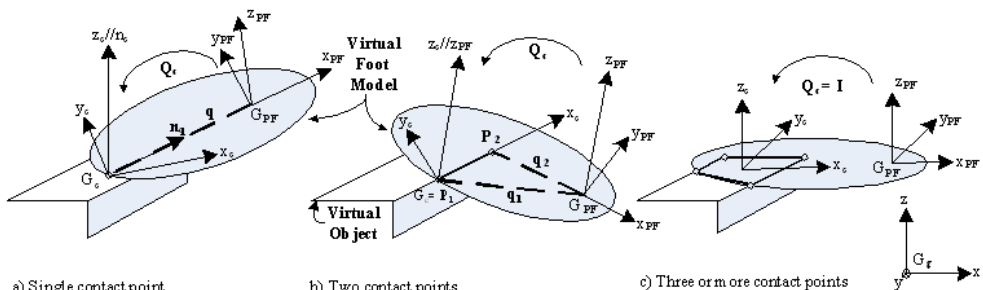

Fig. 7. Contact points description for the three cases

The theory, in the following, applies only for contacts with relatively low deformation. When the deformation is non-linear, alternative methods must be used. In the particular case of a linear deformation, there are three possibilities for which the constraints must be evaluated: the case of a single contact point (section 4.1), two contact points (section 4.2), and three or more contact points (section 4.3) when the wrench $\mathbf{h}_a$ is in the same direction as the normal vector describing contact points geometry.

4.1 Single contact point

The presence of a single contact point is a special case where the contact detection algorithm of the physics engine only finds points that are all situated within a minimal distance, and thus do not generate enough supporting action to some DOFs of the platform that would otherwise have been constrained. This case therefore only constrains the platform in the direction of the normal vector $\mathbf{n}_c$ defined by the tangent plane of the virtual object's surface at the contact point, assuming that friction vectors lie in this plane; the other directions are left unconstrained, i.e. free to move around, as shown in figure 7a). Thus, $\mathbf{S}_c[2][2]$ is set to one and all other values are set to zero, since the z_c axis is set in the normal direction of the contact point.

It must be noted that the determination of rotation matrix $\mathbf{Q}_c$ is difficult because only one z_c axis is defined. An alternative way to compute the force in the contact points reference frame G_c is to first compute $\mathbf{n}_c$, $\mathbf{h}_a$ and $\mathbf{q}$ in the global reference frame G_g, and then find the projection of $\mathbf{h}_a$ according to (10) instead of using the regular FOP (there is no force optimization on one contact point and $\mathbf{Q}_c$ is unknown):

$$\text{if}(\mathbf{q}^T \mathbf{h}_a[0:2] \geq 0) \ \Rightarrow \mathbf{\Gamma}_a = \mathbf{n}_q(\mathbf{n}_q^T \mathbf{h}_a[0:2]) + \tag{10}$$
$$\text{skew}(\mathbf{q})^{-I} \mathbf{h}_a[3:5]$$

$$\text{else } \mathbf{\Gamma}_a = \mathbf{0}_{3 \times 1} \ , \text{ with} \tag{11}$$
$$\mathbf{n}_q = \mathbf{q}/\|\mathbf{q}\|.$$

where [0:2] and [3:5] are operators that select the force and the torque vectors respectively and the *skew()* operator gives a square skew-symmetric matrix.

4.2 Two contact points

In the case of two contact points, the platform has only one free DOF left, as shown in figure 7b). The rotation around the x_c axis is constrained in impedance (null force control) while the other DOF can be controlled in admittance for a rigid virtual object. Rotation matrix $\mathbf{Q}_c$ is computed with the z_c axis parallel to z_{PF} and the x_c axis in the direction of the line linking the two contact points. This rotation matrix is thus defined by (12):

$$\mathbf{Q}_c = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \ , \text{ with} \tag{12}$$

$$\text{if}(P_{2x} - P_{1x} \neq 0) \ \Rightarrow \phi = \arctan \frac{P_{2y} - P_{1y}}{P_{2x} - P_{1x}}, \tag{13}$$

$$\text{else } \phi = \pi/2 \ . \tag{14}$$

The diagonal of the selection matrix $\mathbf{S}_c$ is set so that only linear movements along the x_c and y_c axis with rotations around z_c can be controlled in impedance so as to allow friction forces to be applied, and such that linear movement along the z_c axis and rotation around the y_c axis are constrained in admittance for a rigid virtual object. Only the component representing rotations around the x_c axis in $\mathbf{S}_o$ is set to zero while all other values on the diagonal are set to one in order to select null force control.

4.3 Three or more contact points
This situation is simple because all haptic foot platform DOFs are constrained when some components of $\mathbf{h}_a$ push on the virtual object. Thus, rotation matrix $\mathbf{Q}_c$ and selection matrix $\mathbf{S}_o$ become identity matrices (figure 7c)) and the components of the diagonal of $\mathbf{S}_c$ are set to one except for the components representing linear movement along x_c and y_c axis that are set to zero so as to allow friction effects using impedance control.

5. Haptic Display Rendering (HDR)

To simulate soft objects, the collision detection algorithm from the Newton Game Dynamics™ engine is employed in conjunction with a custom physics engine, labeled HDR, based on the H3D API architecture and some algorithms in ODE (Open Dynamic Engine) optimized for the multi-contact points approach. This section describes the HDR in detail so as to be compatible with cable-driven locomotion interface applications and with the desired hybrid control scheme including wrench sensors designed to obtain the best possible haptic display.

The HDR developed in this paper is based on (Boyd & Wegbreit, 2007) simulation systems combined with (Ruspini & Khatib, 2000) definition of contact space. The solution to the force optimization problem, presented in section 5.2, which is computationally intensive, was proposed in (Baraff, 1994), (Cheng & Orin, 1990) and (Boyd & Wegbreit, 2007). The approach presented in this section assumes that an object is linearly deformable with respect to an impedance model as described in (Ramanathan & Metaxas, 2000) that include a static or dynamic proxy (Mitra & Niemeyer, 2005) and a friction cone law. Force display rendering can be done by other known engines like Chai3d[L]. As a secondary engine, Newton Game Dynamics, embedded in the virtual environment manager, has been chosen among others to provide force feedback of rigid body and collision detection algorithm.

---

[1]http://www.chai3d.org/

5.1 Computation of the Reaction Wrench

The computation of the reaction wrench $\mathbf{h_r}$ employs the action wrench $\mathbf{h}_a$ measured with the 6DOF force/torque sensors placed under the foot in the platform coordinates at origin position G_{PF}. Note that $\mathbf{h}_a$ is defined as the wrench applied by the walker on a haptic foot platform as described in figure 8 and $\mathbf{h}_r$ results from the impedance model of a virtual object and the friction model computed by equation (15):

$$\mathbf{h_r} = \left[\begin{array}{c} \sum_{i=0}^{m-1} \mathbf{Q}_c \mathbf{\Gamma}_{r_i} \\ \sum_{i=0}^{m-1} (\mathbf{q}_i \times (\mathbf{Q}_c \mathbf{\Gamma}_{r_i})) \end{array} \right]_, \tag{15}$$

where $\mathbf{\Gamma}_{ri} \in \mathbb{R}^3$ is the reaction force at the ith contact point $\mathbf{q}_i \in \mathbb{R}^3$. Although the presented algorithms can take into account an arbitrary number of contact points m, the demonstration and results uses only four points, for visual representation, around each rectangular prism that serves as a foot bounding box.

During a collision, each contact point must satisfy four constraints, which are defined similarly to what is presented in (Baraff, 1994):

1. $\mathbf{\Gamma}_{ri}$ can allow penetration between a virtual foot model and a virtual object;
2. $\mathbf{\Gamma}_{ri}$ can push but not pull (there is no glue on the virtual object);
3. $\mathbf{\Gamma}_{ri}$ occurs only at contact points defined on a virtual foot model bounding box, and
4. there is no torque on any point $\mathbf{q}_i$; the reaction torque applied on the virtual foot model is computed by $\mathbf{q}_i \times \mathbf{\Gamma}_{ri}$ as in equation (15).

The reaction forces $\mathbf{\Gamma}_{ri}$ (equation (16)) are composed of the friction forces $\mathbf{\Gamma}_{fi}$ described by the Coulomb law model (equation (19) under constraints (18)), the impedance models $\mathbf{\Gamma}_{Ii}$ (equation (17)), and a given forces $\mathbf{\Gamma}_{Mi}$ whose purpose are to ensure the conservation of linear momentum with a desired restitution coefficient (not presented in this paper):

$$\mathbf{\Gamma}_{r_i} = \mathbf{\Gamma}_{I_i} + \mathbf{\Gamma}_{f_i} + \mathbf{\Gamma}_{M_i} \text{ with,} \tag{16}$$

$$\mathbf{\Gamma}_{I_i} = \mathbf{A}_i \ddot{\mathbf{b}}_i + \mathbf{B}_i \dot{\mathbf{b}}_i + \mathbf{K}_i \mathbf{b}_i, \tag{17}$$

$$\text{if } (\mu_c \mathbf{n}_{ci}^T \mathbf{\Gamma}_{a_i} \leq \|(\mathbf{I} - \mathbf{n}_{ci} \mathbf{n}_{ci}^T) \mathbf{\Gamma}_{a_i}\|) \Rightarrow \tag{18}$$

$$\mathbf{\Gamma}_{f_i} = (-\mu_c \mathbf{n}_{ci}^T \mathbf{\Gamma}_{ai}) \mathbf{t}_{ci}$$
$$\text{else } \mathbf{\Gamma}_{f_i} = -(\mathbf{I} - \mathbf{n}_{ci} \mathbf{n}_{ci}^T) \mathbf{\Gamma}_{a_i}, \tag{19}$$

where $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{K}_i$ are respectively the inertia matrices, the damping matrices and the spring matrices for given penetrations $\mathbf{b}_i$ of a virtual foot model inside a virtual object as shown in figure 9, for small displacements and for linear elasticities, since the contact model

assumes the absence of coupling between each contact point. μ_c is the dynamic friction coefficient, while $\mathbf{n}_{ci}$ and $\mathbf{t}_{ci}$ are the normal and tangential vectors at the interface of a contact point between the virtual foot model and a colliding virtual object computed by the Newton engine and dynamic proxy objects.
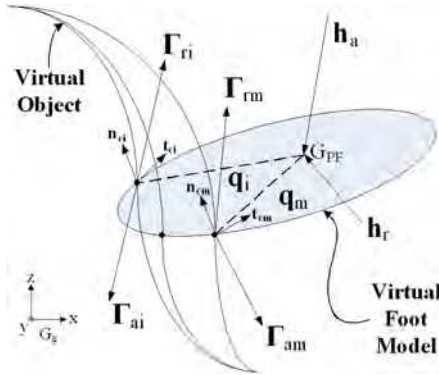

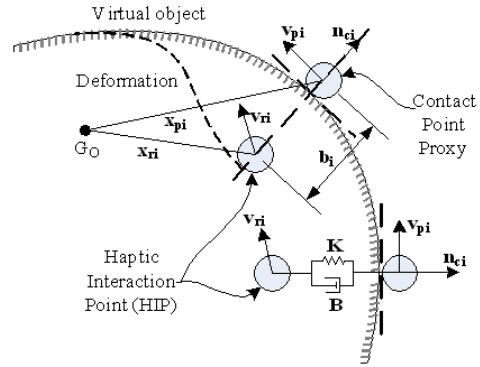Fig. 8. Collision model with action and reaction wrenches

Fig. 9. Contact point proxy for each contact points with the respective penetration

5.2 Force Optimization Problem (FOP)

This section presents the methodology for computing the action forces Γ_{ai} at each contact point under friction cone constraints using the force optimization problem (FOP). The action wrench is measured in the platform reference frame at the location of the 6DOF sensor (G_{PF}). It must then be transferred to each contact point of the foot bounding box in order to obtain the desired virtual representation of the user-applied force. Because no model that calls for a specific force distribution under the foot is used, the action wrench is simply distributed uniformly and optimally, as described in (Duriez, et al. 2006). It is worth noting that this distribution should be evaluated by a walkway sensor array as specified in (Reilly, et al. 1991), but such a sensor has not yet been implemented in this work.

The FOP involves two constraints: the equilibrium constraint and the friction cone constraint, similar to (Melder & Harwin, 2004). The former constraint type is defined by a set of m linear equations (20), with contact matrices $\mathbf{R} \in \mathbb{R}^{6 \times 3m}$ being defined by equations (21) and (22), where $\Gamma_{ai} \in \mathbb{R}^3$ is the ith optimal force used to construct vector $\Gamma_a = [\Gamma_{a0} \dots \Gamma_{a(m-1)}]$:

$$\Pi_o \mathbf{h}_a = \Pi_o \mathbf{R} \Gamma_a \text{ with,} \tag{20}$$

$$\mathbf{R} = [\mathbf{R}_0 \dots \mathbf{R}_{m-1}], \tag{21}$$

$$\mathbf{R}_i = \begin{bmatrix} \mathbf{Q}_c \\ \mathbf{q}_i \times \mathbf{Q}_c \end{bmatrix} . \tag{22}$$

Friction cone constraints are used to define the friction force threshold values at which the virtual foot model transitions between slipping and sticking on an object surface occur. The FOP then attempts to compute the optimal forces when the virtual foot model sticks to the object, and assumes slipping motion when no solution can be found. Hence, the formulation of the FOP can be implemented using quadratic programming with non-linear constraints as represented by equation (23) for any $m \in N^+$:

$$
\begin{cases}
\text{minimize} & \frac{1}{2}\boldsymbol{\Gamma}_a^T \mathbf{H} \boldsymbol{\Gamma}_a \\
& \boldsymbol{\Pi}_o \mathbf{h}_a = \boldsymbol{\Pi}_o \mathbf{R} \boldsymbol{\Gamma}_a \\
\text{under} & \frac{-\boldsymbol{\Gamma}_{a_i}^T \mathbf{n}_{ci}}{\|\boldsymbol{\Gamma}_{a_i}\|} \geq \cos\alpha_i \\
& \text{If no solution, try with} \\
& \mu_c \mathbf{n}_{ci}^T \boldsymbol{\Gamma}_{a_i} \leq \|(\mathbf{I} - \mathbf{n}_{ci}\mathbf{n}_{ci}^T)\boldsymbol{\Gamma}_{a_i}\| \\
i = 0 \dots m-1 \\
\tan\alpha_i = \mu_s \\
H = \text{diag}(h_i),
\end{cases} \tag{23}
$$

where $\mathbf{H}$ is a weighting matrix with $h_i = 1$ which could represent the force distribution under the foot (unused for this work) and μ_s is the static friction coefficient.

5.3. Results for the FOP

This section presents results obtained from the HDR algorithm and its hybrid control strategy. For demonstration purposes, only the four points at the four vertices of the rectangular prism representing a virtual foot model bounding box are used. Note that the number of contact points has to be chosen so as to account for the maximum allowed communication bandwidth between the VEM and the *controller manager*. Figures 10 and 11 show the actual scaled version of the CDLI with a Kondo KHR-1HV.

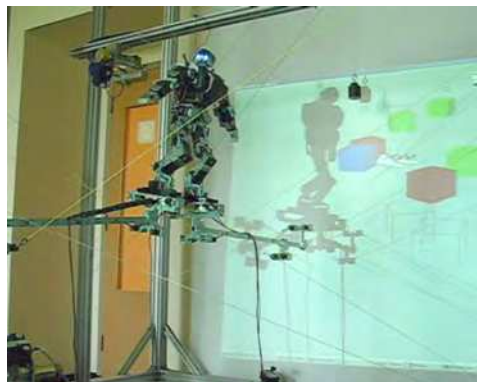Fig. 10. Feet of the Kondo KHR-1HV on the scaled version of the CDLI

Fig. 11. Full view of the scaled version of the CDLI showing the platforms, the cables and the Virtual Reality screen displaying the scene

The simulation parameters are derived from a straight normal walking trajectory described in (McFadyen & Prince, 2002) with its corresponding wrench data defined over six DOFs for a walker mass of about 67 kg. The data consists of force and torque measurements that are collected at a sampling rate of 100 Hz, when the user walks on a rigid floor during a single gait cycle, as seen in figure 10.
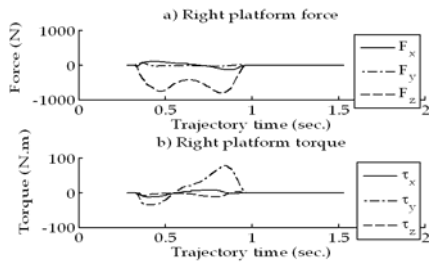

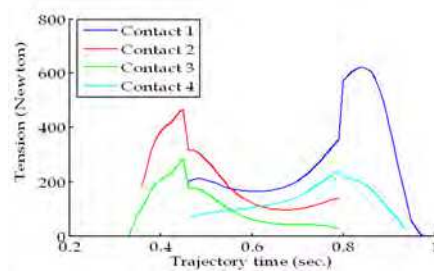Fig. 12. Cartesian reaction wrench applied on the right haptic foot platform

Fig. 13. Normalizes sum of reaction forces $\|\mathbf{\Gamma}_{ri}\|$ at each contact point

The forces generated at each contact point result from the contact points geometry under the virtual foot model and the action wrench, which partly explains why increasing the number of contact points enhances some contact force discontinuities that occasionally occur for a given wrench. Note that this type of discontinuity is expected since the system being optimized in the equation (20) changes its configuration. Figure 13 shows these discontinuities for a right foot trajectory that is subject to (16). Attempts to eliminate these discontinuities *a posteriori* is cumbersome and quite useless since they will be reduced and/or eliminated as the virtual foot model increases in complexity, thereby resulting in a contact distribution that better represents reality.

However, discontinuities in wrench $\mathbf{h}_o$ are still prohibited as they can potentially generate cable tension discontinuities when using the Optimal Tension Distribution (OTD) algorithm in conjunction with the cable tension controllers. When such discontinuities occur, the cable tension controllers cannot follow the computed cable tensions, and the resulting wrench applied on the haptic foot platform can then become unbalanced. Other stability problems are presented in (Joly & Micaelli, 1998).

Note that the presence of only four contact points per virtual foot model is advantageous for visual representation of force distributions, as shown in figure 16, which represents the frames of the video sequence extracted from the HDR and FOP algorithms over one walking gait cycle.

While a reaction force is applied to a haptic foot platform during impedance or admittance control, the action wrench $\mathbf{h}_a$ measured under the foot is employed by the FOP algorithm to compute friction forces at each contact point. The conditions represented by the friction cone are plotted in figure 14, and imply that some contact points slip on the virtual object when the tension forces go below $\cos(\alpha_i)$, thus indicating that a friction force, shown in figure 15, must be added as a reaction force at these points.
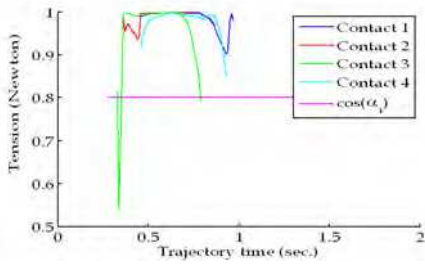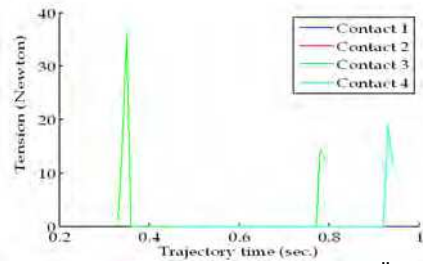
Fig. 14. Friction cone condition

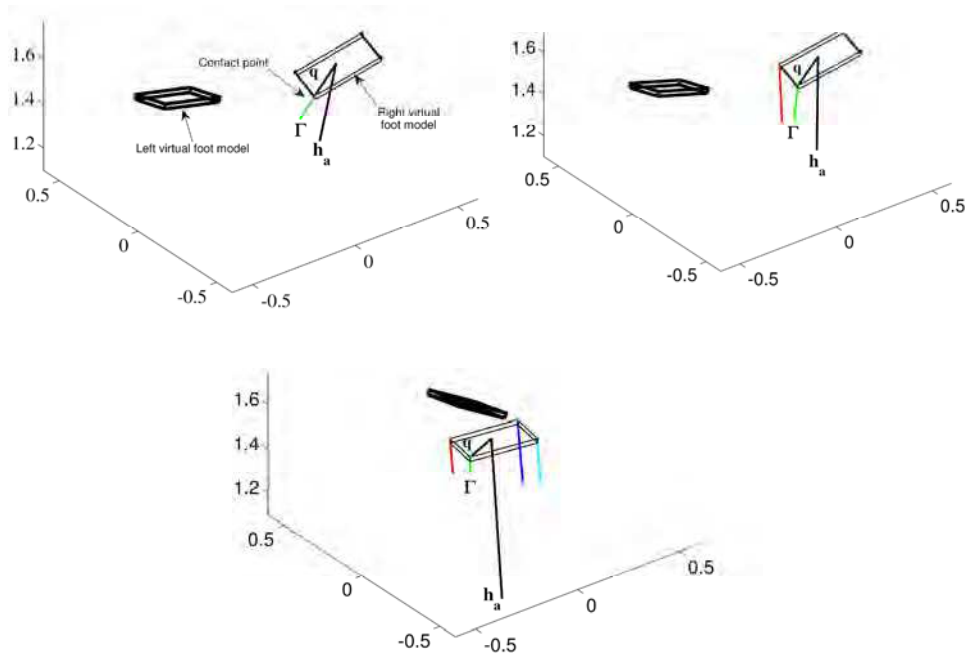Fig. 15. Norm of the friction force $\|\mathbf{\Gamma}_{fi}\|$ as a part of reaction force

Fig. 16. Sequence of the walking simulation with four contact points

6. High dynamic impacts

The CDLI and the FOP presented in the preceding section were developed to render a haptic force feedback that was meant to stimulate the human kinesthetic sense. This sense is what gives humans the perception of force in their muscles. It is of course highly solicited during normal human gait, namely because of the reaction force that the ground inflicts on the foot which is also felt throughout the leg. There is however another sense that is neglected by this mechanism as well as by many other haptic mechanisms. This other sense is called the tactile sense and it is caused by tiny mechanoreceptors situated in glabrous

skin. Some of these receptors are specialized in measuring the strength of deformation of the skin and others are specialized in measuring the changes in deformation of the skin. With this sense, a person is therefore able to feel a material's texture by pressing his or her skin on it's surface and is also able to determine an object's hardness and rigidity upon making contact. The former sensation is not within the scope of the present research and will therefore not be discussed any further. The latter sensation is the one that is most important to this research and it is caused by the transient vibration patterns that occur during a contact (more so during an impact) that are perceivable by these mechanoreceptors within human skin. Since different materials produce different vibration patterns, a person is therefore able differentiate between various materials (Westling and Johanson, 1987). If this sensation could be implemented in the CDLI, a walker could potentially be able to know which material constitutes the virtual floor on which he or she is walking.

The motorized reels presented in (Otis et al. 2009b) that are used in the CDLI were designed mainly to stimulate the human kinesthetic sense. In other words, they were designed to produce a wrench upon the user. These reels, shown in figure 17, are equipped with a transmission and for that reason they are also equipped with a cable tension sensor. In this way, tension control can be achieved via a closed-loop control method.
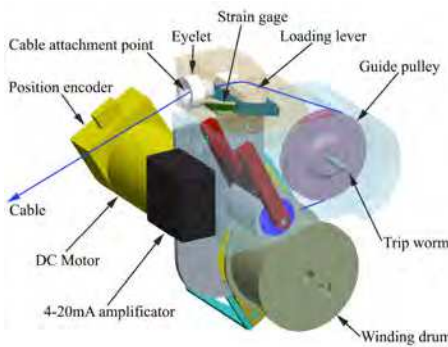

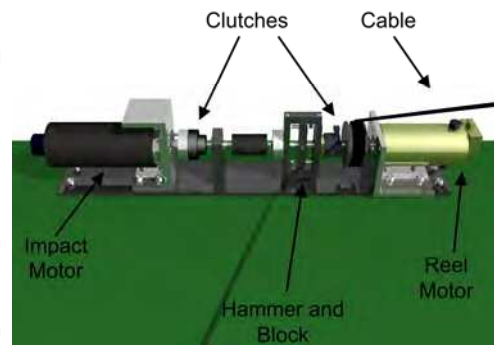Fig. 17. First reel design   Fig. 18. Impact generating reel with two motors

A potential substitute for the previously mentioned reel is shown in figure 18. It was presented for the first time in (Billette and Gosselin, 2009) as a means of producing rigid contacts in simulations such as sword fighting simulations. It was designed to not only be able to stimulate the user's kinesthetic sense but also his tactile sense. To accomplish the latter with conventional reels would be quite hard given the fact that in order to stimulate the mechanoreceptors, they would need to create vibrations with frequencies much higher than 100 Hz. Evidently, if someone were to try and obtain such vibration frequencies with a standard electrical motor and reel he would be faced with the following conundrum: If he minimizes the mechanism's inertia enough to be able to reach these frequencies, the mechanism will not be strong enough to produce the required torque. The prototype in figure 18 addresses this issue by completely rethinking the contact strategy. Instead of trying to simulate impacts, this reel simply produces them by colliding two metal parts.

It takes just one quick look at the prototype reel to see that there is nothing standard about it. The most important parts are the hammer and the block. These are the actual metal parts that will collide during a contact. Since the block is attached permanently to the reel, it

allows the transient vibrations to travel across the cable to the end-effector and the user. The other elements worth noticing are the fact that there are actually two motors instead of one and there are also two clutches added to the system. On the right side, there is the reel motor whose function is to keep tension in the cable at all times. The motor on the left side, called the impact motor, is the motor that will provide the energy for the impacts. The purpose of the two clutches is to control the angular spacing between the hammer and the block. Whenever the mechanism is in "no-contact" mode, the clutches make the two metal parts move together. The hammer is kept at a ready position in a similar manner with which the hammer of a firearm is cocked when ready to fire. In this mode, the impact motor is kept separated from the rest of the reel and the hammer and block assembly turns with the reel motor. When a contact (or impact) is ordered and generated, the clutches change states and this enables the impact motor to grab a hold of the hammer which then becomes free to move with respect to the block. The impact motor moves the hammer with an angular velocity that corresponds to the velocity of the virtual object and the block's movement corresponds to the velocity of the end-effector held by the user. The two metal parts will then collide and generate the required vibrations.

The challenge with the impact generation strategy described above comes from the fact that the vibrations must travel across all of the cables. Parallel cable driven mechanisms have typically small rigidity compared to solid member parallel mechanisms and it is therefore safe to assume that these vibrations will be dampened and that the highest vibrations frequencies generated at the reel may not travel across the cables. However, preliminary tests have shown that although the transient vibration patterns do not resemble those that would have occurred if the end-effector were to strike a real steel object, they do however show a close resemblance to the patterns of a material that can be considered as moderately rigid and hard (delrin). Applied to the CDLI, these reels could potentially give the walker an improved walking sensation by providing a punctuality to the reaction forces that he feels upon setting his foot on the virtual ground. Also, such reaction forces could also increase the haptic rendering for other activities such as striking a movable virtual object with a foot.

7. Conclusion

The haptic mechanism exploits software and hardware architectures that were specifically designed for managing a Cable-Driven Locomotion Interface driven by a haptic rendering engine for real-time applications. The architecture includes hybrid impedance, admittance and inertial-wrench control classes and two physics engines that permits the best haptic display for soft and rigid virtual objects. These components are implemented and generalized following an open-architecture paradigm in order to render a haptic display, and for facilitating physical model implementation.

The core of the control class selection mechanism is a selection matrix that depends on both the contact points geometry and the virtual object physical properties. Such a mechanism selects a particular control scheme for each haptic foot platform DOF, depending on the type of collision and friction conditions. The Force Optimization Problem then only needs to be solved over this spatial geometry, and is constrained by a friction cone which can be computed using non-linear quadratic programming algorithms. However, not only a standard reel design but also the cable-driven mechanism can not support high impact

dynamics. Further investigation is needed for controlling vibrations that could occur between two rigid contacts.

8. Future work

The current model for the simulation of soft virtual objects is still under development. The coupling between each contact point is currently being neglected, and equation (12) is only valid for small penetrations and for linear elasticity tensors. It is possible to extend the friction model with more complex algorithms in order to consider nonlinearities like Signorini's law implemented in (Duriez et al., 2006). Haptic synthesis of interaction with novel materials (e.g., soil, sand, water, stone) with non-linear deformation and multimodal (audio and haptic) rendering will need to be developed for increasing realism. Such synthesis needs novel sensor network design for distributed interactive floor surfaces.

Concerning the locomotion interface, a washout filter with force feedback that uses an impedance model will be implemented to continuously drive the user toward the centre of the workspace. As for the haptic display accuracy, it can be increased by analyzing the real force distribution under a human foot.

Acknowledgment

9. References

Adams, R.J. & Hannaford, B. (1999). Stable haptic interaction with virtual environments. *IEEE Transactions on Robotics and Automation*, Vol. 15, No. 3, June 1999, pp. 465 – 74, ISSN 1042-296X.

Adams, R.J.; Klowden, D. & Hannaford, B. (2000). Stable haptic interaction using the Excalibur force display, *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 770-775, ISBN-10: 0 78035 886 4, San Francisco, CA, USA, April 24-28, 2000, IEEE Robotics and Automation Society, Piscataway, NJ, USA

Baraff., D. (1994). Fast contact force computation for nonpenetrating rigid bodies, *Proceedings of Conference on Computer graphics and interactive techniques (SIGGRAPH)*, pp. 23 – 34, ISBN-10: 0 89791 667 0, Orlando, FL, USA, July 1994, ACM Press, New York, NY, USA

Barrette, G. & Gosselin, C. (2005) Determination of the dynamic workspace of cable-driven planar parallel mechanisms. *Journal of Mechanical Design, Transactions of the ASME*, Vol. 127, No. 2, March 2005, pp. 242 – 248, ISSN 0738-0666.

Bernhardt, M.; Frey, M.; Colombo, G. & Riener, R. (2005). Hybrid force-position control yields cooperative behaviour of the rehabilitation robot lokomat, *Proceedings of IEEE International Conference on Rehabilitation Robotics*, pp. 536 – 539, ISBN-10: 0780390032, Chicago, IL, USA, June-July 2005, IEEE Computer Society, Piscataway, NJ, USA.

Billette, G. & Gosselin, C. (2009). Producing Rigid Contacts in Cable-Driven Haptic Interfaces Using Impact Generating Reels, *Proceedings of International Conference on Robotics and Automation*, pp. 307-312, ISBN-13 9781424427888 , 2009, Kobe, Japan, IEEE, Piscataway, NJ, USA.

Boyd, S. & Wegbreit, B. (2007). Fast computation of optimal contact forces. *IEEE Transactions on Robotics*, Vol. 23, No. 6, December 2007, pp. 1117 – 1132, ISSN 1552-3098.

Carignan, C.; & Cleary, K. (2000). Closed-loop force control for haptic simulation of virtual environments. *Haptics-e*, Vol. 1, No. 2, February 2000, pp. 1 – 14.

Changhyun Cho; Jae-Bok Song & Munsang Kim (2008). Stable haptic display of slowly updated virtual environment with multirate wave transform. *IEEE/ASME Transactions on Mechatronics*, Vol. 13, No. 5, pp. 566 – 575, ISSN 1083-4435.

Cheah, C.C.; Kawamura, S. & Arimoto, S. (2003). Stability of hybrid position and force control for robotic manipulator with kinematics and dynamics uncertainties. *Automatica*, Vol. 39, No. 5, May 2003, pp. 847-855, ISSN 0005-1098.

Cheng, F.-T. & Orin, D. E. (1990). Efficient algorithm for optimal force distribution - the compact-dual lp method. *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 2, April 1990, pp. 178 – 187, ISSN: 1042-296X.

Duchaine, V. & Gosselin, C. (2007). General model of human-robot cooperation using a novel velocity based variable impedance control, *Proceedings of EuroHaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, pp. 446–451, ISBN-10 0769527388, Tsukaba, Japan, March 2007, IEEE Computer Society, Los Alamitos, CA, USA.

Duchaine, V. & Gosselin, C. (2009). Safe, Stable and Intuitive Control for Physical Human-Robot Interaction, *Proceedings of International Conference on Robotics and Automation*, pp. 3383-3388, ISBN-13 9781424427888, Kobe, Japan, May 12-17, 2009, IEEE, Piscataway, NJ, USA.

Duriez, C.; Dubois, F.; Kheddar, A. & Andriot, C. (2006). Realistic haptic rendering of interacting deformable objects in virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 12, No. 1, January 2006 , pp. 36 – 47, ISSN 1077-2626.

Faulring, E.L.; Lynch, K.M.; Colgate, J.E. & Peshkin, M.A. (2007). Haptic display of constrained dynamic systems via admittance displays. *IEEE Transactions on Robotics*, Vol. 23, No. 1, February 2007, pp. 101-111, ISSN 1552-3098.

Fang , S.; Franitza D.; Torlo M.; Bekes, F. & Hiller, M. (2004). Motion control of a tendon-based parallel manipulator using optimal tension distribution. *IEEE/ASME Transactions on Mechatronics*, Vol. 9, No. 3, September 2004, pp. 561– 568, ISSN 1083-4435.

Goldsmith, P.B.; Francis, B.A.; Goldenberg, A.A. (1999). Stability of hybrid position/force control applied to manipulators with flexible joints. *International Journal of Robotics & Automation*, Vol. 14, No. 4, 1999, pp. 146-160, ISSN 0826-8185.

Grow, David I. & Hollerback, John M. (2006). Harness design and coupling stiffness for two-axis torso haptics, *International Conference on IEEE Virtual Reality*, pp. 83-87, ISBN 1424402263, Alexandria, VA, United states, 25-26 March 2006, Piscataway, NJ, USA.

Hannaford, B. & Ryu, J.-H. (2002). Time-domain passivity control of haptic interfaces. *IEEE Transactions on Robotics and Automation*, Vol. 18, No 1, February 2002, pp. 1-10, ISSN 1042-296X.

Hassan, M. & Khajepour, A. (2007). Optimization of actuator forces in cable-based parallel manipulators using convex analysis. *IEEE Transactions on Robotics*, Vol. 24, No. 3, June 2008, pp. 736 - 740, ISSN 15523098

Iwata, H.; Yano, H. & Nakaizumi, F. (2001). Gait master: a versatile locomotion interface for uneven virtual terrain, *Proceedings of IEEE Virtual Reality*, pp. 131 – 137, ISBN-10 0769509487, Yokohama, Japan, March 2001, IEEE Computer Society, Los Alamitos, CA, USA.

Joly, L. & Micaelli, A. (1998). Hybrid position/force control, velocity projection, and passivity, *Proceedings of Symposium on Robot Control (SYROCO)*, Vol. 1, pp. 325 – 331, ISBN-10 0080430260, Nantes, France, September 1997, Elsevier, Kidlington, UK.

Kawamura, S.; Ida, M; Wada, T. & Wu, J.-L. (1995). Development of a virtual sports machine using a wire drive system-a trial of virtual tennis, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, Human Robot Interaction and Cooperative Robots*, Vol. 1, pp. 111 – 116, Pittsburgh, PA, USA, August 1995, IEEE Computer Society, Los Alamitos, CA, USA.

Lauzier, N.; Gosselin, C. (2009). 2 DOF Cartesian Force Limiting Device for Safe Physical Human-Robot Interaction, *Proceedings of International Conference on Robotics and Automation*, pp. 253-258, ISBN-13 9781424427888, Kobe, Japon, 12-17 May 2009, IEEE, Piscataway, NJ, USA.

Lu , X.; Song, A. (2008). Stable haptic rendering with detailed energy-compensating control. Computers & Graphics, Vol. 32, No. 5, October 2008, pp. 561-567, ISSN 0097-8493.

McFadyen, B. J. & Prince, F. (2002). Avoidance and accomodation of surface height changes by healty, community-dwelling, young, and elderly men. *Journal of Gerontology: Biological sciences*, Vol. 57A, No. 4, April 2002, pp. B166–B174, ISSN 1079-5006.

McJunkin, S.T.; O'Malley, M.K. & Speich, J.E. (2005). Transparency of a Phantom premium haptic interface for active and passive human interaction, *Proceedings of the American Control Conference*, pp. 3060 – 3065, ISBN-10 0 7803 9098 9, Portland, OR, USA, 8-10 June, 2005, IEEE, Piscataway, NJ, USA.

Melder, N. & Harwin, W. (2004). Extending the friction cone algorithm for arbitrary polygon based haptic objects, *Proceedings of International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems (HAPTICS)*, pp. 234 – 241, ISBN-10 0769521126, Chicago, IL, United States, March 2004, IEEE Computer Society, Los Alamitos, CA, USA.

Mitra, P. & Niemeyer, G. (2005). Dynamic proxy objects in haptic simulations, *Proceedings of Conference on Robotics, Automation and Mechatronics*, Vol. 2, pp. 1054 – 1059, ISBN-10 0780386450, Singapore, IEEE, Piscataway, NJ, USA.

Morizono, T.; Kurahashi, K. & Kawamura, S. (1997). Realization of a virtual sports training system with parallel wire mechanism, *Proceedings of IEEE International Conference on Robotics and Automation*, Vol. 4, pp. 3025 – 3030, ISBN-10 0780336127, Albuquerque, NM, USA, April 1997, IEEE Robotic and Automation Society, New York, NY, USA.

Onuki, K.; Yano, H.; Saitou, H. & Iwata, H. (2007). Gait rehabilitation with a movable locomotion interface. *Transactions of the Society of Instrument and Control Engineers*, Vol. 43, No. 3, 2007, pp. 189 – 196, ISSN 0453-4654.

Otis, M. J.-D.; Perreault, S.; Nguyen-Dang, T.-L.; Lambert, P.; Gouttefarde, M.; Laurendeau, D.; Gosselin, C. (2009a). Determination and Management of Cable Interferences Between Two 6-DOF Foot Platforms in a Cable-Driven Locomotion Interface. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, Vol. 39, No. 3, May 2009, pp. 528-544, ISSN 1083-4427.

Otis, M. J.-D.; Nguyen-Dang, T.-L.; Laliberte, Thierry; Ouellet, Denis; Laurendeau, D.; Gosselin, C. (2009b). Cable Tension Control and Analysis of Reel Transparency for 6-DOF Haptic Foot Platform on a Cable-Driven Locomotion Interface. *International Journal of Electrical, Computer, and Systems Engineering*, Vol. 3, No. 1, May 2009, pp. 16-29, ISSN 2070-3813.

Ottaviano, E.; Castelli, G.; Cannella, G. (2008). A cable-based system for aiding elderly people in sit-to-stand transfer. *Mechanics Based Design of Structures and Machines*, Vol. 36, No. 4, October 2008, pp. 310 – 329, ISSN 1539-7734.

Perreault, S. & Gosselin, C. (2008). Cable-driven parallel mechanisms: application to a locomotion interface. *Journal of Mechanical Design, Transactions of the ASME*, Vol. 130, No. 10, October 2008, pp. 102301-1-8, ISSN 0738-0666.

Ramanathan, R. & Metaxas, D. (2000). Dynamic deformable models for enhanced haptic rendering in virtual environments, *Proceedings of Virtual Reality Annual International Symposium*, pp. 31 – 35, ISBN-10 0769504787, New Brunswick, NJ, USA, March 2000, IEEE Computer Society, Los Alamitos, CA, USA.

Reilly, R., Amirinia, M. & Soames, R. (1991). A two-dimensional imaging walkway for gait analysis, *Proceedings of Computer-Based Medical Systems Symposium*, pp. 145 – 52, ISBN-10 0818621648, Baltimore, MD, USA, May 1991, IEEE Computer Society, Los Alamitos, CA, USA.

Ruspini, D. & Khatib, O. (2000). A framework for multi-contact multi-body dynamic simulation and haptic display, *Proceedings of International Conference on Intelligent Robots and Systems*, Vol. 2, pp. 1322 – 1327, ISBN-10 0780363485, Takamatsu, Japon, November 2000, IEEE, Piscataway, NJ, USA.

Sakr, N.; Jilin, Z.; Georganas, N.D; Jiying Z. & Petriu, E.M. (2009). Robust perception-based data reduction and transmission in telehaptic systems, *Proceedings of World Haptics Conference*, pp. 214-219, ISBN-13 9781424438587, Salt Lake City, UT, USA, March 2009, IEEE, Piscataway, NJ, USA.

Schmidt, H.; Hesse, S. & Bernhardt, R. (2005). Hapticwalker - a novel haptic foot device. *ACM Transaction on Applied Perception*, Vol. 2, No. 2., April 2005, pp. 166 – 180, ISSN 1544-3558.

SenseGraphics. H3D Open Source Haptics. http://www.h3dapi.org/.

Smith, R. ODE, Open Dynamics Engine. http://www.ode.org/.

Tsumugiwa, T.; Yokogawa, R. & Hara, K. (2002). Variable impedance control with virtual stiffness for human-robot cooperative peg-in-hole task, *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, Vol. 2, pp. 1075 – 1081, ISBN-10 0780373987, Lausanne, Switzerland, September 2002, IEEE Robotics & Automation Society, Piscataway, NJ, USA.

van der Linde, R.Q. & Lammertse, P. (2003). HapticMaster - a generic force controlled robot for human interaction. *Industrial Robot*, Vol. 30, No. 6, 2003, pp. 515-524, ISSN 0143-991X

Westling, G. & Johansson, R. S. (1987). Responses in glabrous skin mechanoreceptors during precision grip in humans. *Experimental Brain Research*, Vol. 66, No. 1, 1987, pp. 128-140, ISSN 0014-4819.

Yoon, J. & Ryu, J. (2004). Continuous walking over various terrains - a walking control algorithm for a 12-dof locomotion interface, *Proceedings of International Conference Knowledge-Based Intelligent Information and Engineering Systems*, Vol. 1, pp. 210 – 217, ISBN-10 3540233180, Wellington, New Zealand, September 2004, Springer-Verlag, Berlin, Germany.

Yoon, J. & Ryu, J. (2006). A novel locomotion interface with two 6-dof parallel manipulators that allows human walking on various virtual terrains. *International Journal of Robotics Research*, Vol. 25, No. 7, July 2006, pp. 689 – 708, ISSN 02783649.

Yoon, J. & Ryu, J. (2009). A Planar Symmetric Walking Cancellation Algorithm for a Foot–Platform Locomotion Interface. *International Journal of Robotics Research*, *in press*, 19 May 2009, pp. 1 – 21.

Advances in Haptics

Edited by Mehrdad Hosseini Zadeh

Haptic interfaces are divided into two main categories: force feedback and tactile. Force feedback interfaces are used to explore and modify remote/virtual objects in three physical dimensions in applications including computer-aided design, computer-assisted surgery, and computer-aided assembly. Tactile interfaces deal with surface properties such as roughness, smoothness, and temperature. Haptic research is intrinsically multi-disciplinary, incorporating computer science/engineering, control, robotics, psychophysics, and human motor control. By extending the scope of research in haptics, advances can be achieved in existing applications such as computer-aided design (CAD), tele-surgery, rehabilitation, scientific visualization, robot-assisted surgery, authentication, and graphical user interfaces (GUI), to name a few. Advances in Haptics presents a number of recent contributions to the field of haptics. Authors from around the world present the results of their research on various issues in the field of haptics.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Martin J.D. Otis, Vincent Duchaine, Greg Billette, Simon Perreault, Clement Gosselin and Denis Laurendeau (2010). Cartesian Control of a Cable-Driven Haptic Mechanism, Advances in Haptics, Mehrdad Hosseini Zadeh (Ed.), ISBN: 978-953-307-093-3, InTech, Available from: http://www.intechopen.com/books/advances-in-haptics/cartesian-control-of-a-cable-driven-haptic-mechanism

INTECH
open science | open minds